

WHAT IS CLAIMED IS:

1. A method of creating multimedia services in a network comprising the steps of:

- a). assembling graphical language blocks into a service logic script;
- b). installing said service logic script in a service execution environment; and

5 c). translating graphical language blocks into programming language objects
when said service logic script is installed and executed.

2. The method according to claim 1, wherein said service logic script captures all
interactions with a service subscriber, requests network resources on behalf of the service
10 subscriber, provides access to all required data, and prepares information for service
billing purposes.

3. The method according to claim 1, wherein Service control and call control
functions of the graphical objects are mapped into service control and call control
15 functions of the Application Programming Interfaces (APIs), which are exposed in the
Service Execution Environment.

4. The method according to claim 1, wherein dependencies are established between
service control function or call control function of the graphical objects and event

20 notifications sent from Service Execution Environment.

5. The method according to claim 1, wherein next state of the service control or call control function is determined upon its completion or upon receiving an event notification from the Service Execution Environment.

5 6. The method according to claim 3, wherein the application programming interfaces provide access to and control of network resources which are used during service invocation by service subscribers.

7. The method according to claim 1, further comprising the step of:
10 determining if a service requires data and automatically, storing, populating and retrieving data during service instantiation.

8. The method according to claim 7, wherein said data comprises at least one of system data, service data, subscriber data and local data.

15 9. The method according to claim 7, further comprising the step of:
generating service provisioning forms for the entry of required data.

10. The method according to claim 7, further comprising the step of:
20 generating subscriber tuning forms to allow a service subscriber to enter data related to the service subscriber.

11. The method according to claim 1, wherein said graphical language blocks each represent unique service control and call control functions.

12. The method according to claim 11, wherein each block has at least one input or
5 output for passing a token between blocks.

13. The method according to claim 12, wherein a block begins execution when the block receives said token.

10 14. The method according to claim 13, wherein a plurality of blocks may execute simultaneously.

15 15. The method according to claim 1, wherein said graphical language blocks represent complex event driven actions which take place in the network and which are hidden from a user.

16. A service creation system for creating multimedia services, comprising:
a service creation environment for creating a service logic script;
a service execution environment for executing said service logic script; and
20 application programming interfaces between said service creation environment and said service execution environment.

17. The service creation system according to claim 16, wherein said service creation environment comprises:

an editor for providing graphical capability to create a service logic script for a desired service using language graphical objects;

5 a translator for translating language graphical objects into lower level language executable objects;

a data repository for storing service and customer related data necessary for the desired system.

10 18. The service creation system according to claim 16, wherein said service execution environment comprises a service locator, a service instantiator, and a service logic executor.

15 19. The service creation system according to claim 18, wherein said service locator identifies service logic based on a service subscriber identification.

20. The service creation system according to claim 19, wherein said service locator selects a service ID and finds a service logic execution environment in which the logic is stored.

20 21. The service creation system according to claim 18, wherein said service instantiator instantiates invoked service logic with required service and subscriber data so as to be ready to be executed in the service execution environment.

22. The service creation system according to claim 18, wherein said service logic
executor loads executable service logic code, reads and stores service and subscriber data,
provides inter-process communication between various service logic scripts that are being
5 executed concurrently.

23. The service creation system according to claim 17, wherein said graphical
language blocks each represent unique service control and call control functions.

10 24. The service creation system according to claim 23, wherein each block has at least
one input or output for passing a token between blocks.

25. The service creation system according to claim 24, wherein a block begins
execution when the block receives said token.

15 26. The service creation system according to claim 25, wherein a plurality of blocks
may execute simultaneously.

R1.126 27.
28. The service creation system according to claim 17, wherein the desired service is
20 a call follow-me service.

28. The service creation system according to claim 17, wherein the desired service is
a restricted calling service.

29. The service creation system according to claim 17, wherein said language graphical objects represent complex event driven actions which take place in a network and which are hidden from a user.

[illegible]